UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/523,031	02/02/2005	Daniel Shane O'Sullivan	1004437.006US	8713

85775 7590 01/20/2010

Locke Lord Bissell & Liddell LLP
Attn: IP Docketing
Three World Financial Center
New York, NY 10281-2101

EXAMINER	
FAHERTY, COREY S	

ART UNIT	PAPER NUMBER
2183	

NOTIFICATION DATE	DELIVERY MODE
01/20/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ptopatentcommunication@lockelord.com

Office Action Summary	Application No.	Applicant(s)	
	10/523,031	O'SULLIVAN, DANIEL SHANE	
	Examiner	Art Unit	
	Corey Faherty	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 November 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *144,145,155-158,160,164,165,167-172 and 1615* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *144,145,155-158,160,164,165,167-172 and 1615* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This office action is in response to the RCE filed on 11/23/2009.

2. Claims 144-145, 155-158, 160, 164-165, 167-172 and 1615 are pending in the application and have been examined.

3. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 11/23/2009 has been entered.

Claim Objections

4. Claim 144 recites the limitations "the processing resources of the first type being each being defined by a configuration whereby..." and "the processing resources of the second type each being defined by being configured to..." Both of these limitations include vague language specifying that processor resources are "defined by" their configurations. It is not clear if this is somehow different than simply being configured as specified. Appropriate clarification is required. Furthermore, the first limitation includes the grammatically incorrect phrase "being each being defined". Correction is required.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it

> pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

6. Claims 144-145, 155-158, 160, 164-165, 167-172 and 1615 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

7. Claim 144 recites the limitation "the processing resources of the second type each being defined by being configured to only receive and execute instructions that were delegated to them by processing resources of the first type". The examiner can find no support for this limitation in the original teachings of the application. The dependent claims also include this subject matter and therefore also fail to meet the requirements of 35 U.S.C. 112, first paragraph.

8. Claim 1615 recites the limitation "receiving an execution instruction of a different thread at the second IPU; determining that the execution-instruction of the different thread can not be processed by the second IPU but can be processed by the other type elemental processing resource that is shared between the second IPU and the first IPU". The examiner can find no support for this limitation in the original teachings of the application.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are

> such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

11. **Claims 144, 155-156, 164-165 and 167-170** are rejected under 35 U.S.C. 103(a) as being unpatentable over Bonola (U.S. Patent 5,706,514) in view of Bridges et al. (U.S. Patent 6,081,860), referenced from here forward as Bridges.

12. Regarding claim 144, Bonola discloses a method of execution-instruction delegation among elemental processing resources of at least two different types [abstract], comprising:

obtaining an execution instruction, wherein the execution instruction is obtained at one of the elemental processing resources of a first type [col. 3, line 26; a command is encountered by a host, or master, processor], wherein the first type of processing resource is configured to receive a non-delegated instruction from memory that it can execute or delegate for execution elsewhere [col. 3, lines 36-51; when an encountered instruction is of a type that cannot be executed by the host processor, the host processor transfers data to a slave processor's registers so that the slave processor may execute the instruction; col. 3, lines 23-44; if the encountered instruction is not of a type that cannot be executed by the host processor, the host processor continues executing it normally];

determining whether an operation-code within the execution instruction is incapable of being executed by the one elemental processing resource of the first type and thus should be delegated to an other elemental processing resource [col. 3, lines 25-29; it is determined if the instruction is of a type that cannot be executed by the host processor] of a second type [col. 3, lines 36-51; when an encountered instruction is of a type that cannot be executed by the host processor, the host processor transfers data to a slave processor's registers so that the slave processor may execute the instruction], wherein the second type of processing resource is configured to only receive and execute instructions that were delegated to them by processing resources of the first type [col. 3, lines 56; the slave processor executes only when an instruction is delegated to it by the host processor; otherwise the slave processor is suspended];

executing the execution instruction with the processing resource of the first type, if the operation-code within the execution instruction should not be delegated to the other processing resource [col. 3, lines 23-44; if the encountered instruction is not of a type that cannot be executed by the host processor, the host processor continues executing it normally]; and

if the execution instruction should be delegated, routing the execution instruction to the elemental processing resource of the second type that is capable of executing the operation code [col. 3, lines 36-51; when an encountered instruction is of a type that cannot currently be executed by the host processor, the host processor transfers data to a slave processor's registers so that the slave processor may execute the instruction].

Bonola does not explicitly disclose that the system has a plurality of host processors that share a plurality of slave processors. However, Bonola does teach that other arrangements of multiprocessor computer systems could be used for carrying out the invention [col. 7, lines 11-

15]. Bridges discloses such an alternate arrangement for a multiprocessor system having multiple master processors and multiple slave processors [Fig. 1]. Bridges further discloses that multiple master processors issue requests to a slave processor [col. 6, lines 33-36]. Such a system allows for any of the master processors to gain the benefit of having a slave processor, and the use of multiple slave processors allows them to do so in parallel, resulting in improved utilization and overall system throughput. It therefore would have been obvious to a person of ordinary skill in the art at the time the invention was made to have multiple host, or master, processors delegate instruction requests to multiple slave processors in the system of Bonola because Bridges teaches such a multiprocessor system for handling master processor requests.

13. Regarding claim 155, Bonola in view of Bridges discloses the method of claim 144, wherein the operation-code indicates a type of resource on which to execute [Bonola, col. 3, lines 25-30; an instruction specifies a certain mode of processor on which it can executed].

14. Regarding claim 156, Bonola in view of Bridges discloses the method of claim 144, wherein the other processing resource may be the originating processing resource [Bonola, col. 3, lines 23-44; if the encountered instruction is not of a type that cannot currently be executed by the host processor, the host processor continues executing it normally].

15. Regarding claim 164, Bonola in view of Bridges discloses the method of claim 144, wherein a processing resource is an execution-instruction processing cache [Bonola, col. 3, lines 37-42; the slave processor's registers are used to temporarily hold all data that is necessary for the slave processor to perform the execution provided to it by the host processor].

16. Regarding claim 165, Bonola in view of Bridges discloses the method of claim 144, wherein routing occurs through an execution-instruction signal router [Bonola, col. 3, lines 37-

42; the slave processor's registers are used to temporarily hold all data that is necessary for the slave processor to perform the execution provided to it by the host processor; Fig. 1; col. 4, lines 43-45; a host bus is also provided to allow the host processor to communicate with the slave processors].

17. Regarding claims 167, Bonola in view of Bridges discloses the method of claim 144, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions [Bonola, col. 8, lines 55-61; col. 9, lines 64-66; a slave processor may enter a sleep mode while the host processor or other slave processors execute instructions].

18. Regarding claim 168, Bonola in view of Bridges discloses the method of claim 144, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off [Bonola, col. 8, lines 55-61; col. 9, lines 64-66; slave processors may enter a sleep mode in which they do not execute instructions].

19. Regarding claim 169, Bonola in view of Bridges discloses the method of claim 144, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling [Bonola, col. 8, lines 55-61; col. 9, lines 64-66; slave processors may enter a sleep mode in which they do not execute instructions].

20. Regarding claim 170, Bonola in view of Bridges discloses the method of claim 144, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required [Bonola, col. 8, lines 55-61; col. 9, lines 64-66; a slave processor may be removed from sleep mode when it is necessary for executing instructions that the host processor is unable to execute].

21. **Claims 144-145 and 171-172** are rejected under 35 U.S.C. 103(a) as being unpatentable over Butterworth et al. (U.S. Patent 6,907,454), referenced from here forward as Butterworth, in view of Bridges.

22. Regarding claim 144, Butterworth discloses a method of execution-instruction delegation between elemental processing resources of at least two different types [abstract], comprising:

obtaining an execution instruction, wherein the execution instruction is obtained at an elemental processing resource of a first type [col. 2, lines 48-50; a master processor receives a memory access instruction], wherein the first type of processing resource is configured to receive a non-delegated instruction from memory that it can execute or delegate for execution elsewhere [col. 2, lines 48-54; the master processor determines that a fetched instruction is of a memory access type and, if it is, determines that it should be sent to a slave processor];

determining whether an operation-code within the execution instruction should be delegated to an other elemental processing resource of a second type different from the first type [col. 2, lines 48-54; the master processor determines that an instruction is of a memory access type and, if it is, determines that it should be sent to a slave processor], wherein the second type of processing resource is configured to only receive and execute instructions that were delegated to them by processing resources of the first type [col. 2, line 44 – col. 3, line 27; the slave processor is configured to perform only a certain type of operation assigned to it by the master processor];

executing the execution instruction with the elemental processing resource of the first type, if the operation-code within the execution instruction should not be delegated to the other

elemental processing resource [col. 2, lines 44-58; the master processor executes non-memory access instructions normally];

routing the execution instruction to the other elemental processing resource, if the operation-code within the execution instruction is for the other elemental processing resource [col. 2, lines 53-54; for memory access instructions, the master processor writes a request to the slave processor to carry out the instruction].

Butterworth does not explicitly disclose that the system has a plurality of master processors that share a plurality of slave processors. Bridges discloses a multiprocessor system having multiple master processors and multiple slave processors [Fig. 1]. Bridges further discloses that multiple master processors issue requests to a slave processor [col. 6, lines 33-36]. Such a system allows for any of the master processors to gain the benefit of having a slave processor. It therefore would have been obvious to a person of ordinary skill in the art at the time the invention was made to have multiple master processors delegate instruction requests to a slave processor in the system of Butterworth because Bridges teaches such a multiprocessor system for more efficiently handling master processor requests.

23. Regarding claim 145, Butterworth in view of Bridges discloses the method of claim 145, but does not explicitly disclose that the method is completed within a single processing cycle.

The purpose of the Butterworth design is to reduce the processing delay of a memory operation [col. 1, lines 59-65]. Instead of forcing a processor to incur all of the delay necessary for executing a memory operation, the processor instead delegates the operation to a second processor [col. 2, lines 44-58]. If the first processor is able to perform this delegation quickly, it can continue executing other instructions [col. 5, lines 50-51] while the second processor

performs the memory operation. The fewer clock cycles it takes for the first processor to perform the delegation, the more time the first processor will have to execute other instructions while the second processor is performing the memory operation.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to complete the instruction-delegation method disclosed in Butterworth in a single processing cycle because doing so would allow the master processor to begin executing other instructions more quickly and thus execute more code overall while the slave processor is performing a memory operation.

24. Regarding claim 171, Butterworth in view of Bridges discloses the method of claim 144, wherein processing resources are communicatively disposed on a same die [Butterworth, col. 4, lines 36-40; a bridge device for communication between the memory and the slave processor may be integrated into a single chip package with the slave processor].

25. Regarding claim 172, Butterworth in view of Bridges discloses the method of claim 171, wherein an execution-instruction signal router is on the same die with processing resources [Butterworth, col. 4, lines 36-40; a bridge device for communication between the memory and the slave processor may be integrated into a single chip package with the slave processor].

26. **Claims 157-158, 160 and 1615** are rejected under 35 U.S.C. 103(a) as being unpatentable over Bonola in view of Bridges as applied to claim 144 above, and further in view of Mohamed et al. (U.S. Patent 5,978,838), referenced from here forward as Mohamed.

27. Regarding claims 157 and 160, Bonola in view of Bridges discloses does not explicitly disclose that the first type of processing resource is an integer-processing unit nor that the other type of processing resource is a vector processing resource.

Mohamed discloses a system similar to that of Bonola in which two processors interact to efficiently process instructions [col. 1, lines 61-67]. The system includes integer-processing units for executing integer instructions [col. 5, lines 8-15]. The system also includes a vector processor [abstract].

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include an integer-processing unit as a processing resource in the system of Bonola because Mohamed discloses including an integer-processing unit in a multi-processor system [col. 5, lines 8-15] and doing so allows the system to execute integer instructions. Furthermore, without an integer-processing unit, the system of Bonola would be unable to execute integer instructions, greatly limiting the amount of software that it could execute.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include a vector-processing unit as a processing resource in the system of Bonola because Mohamed discloses including a vector processor in a multi-processor system [abstract] and doing so allows a system to more efficiently execute vector instructions without impacting the execution efficiency of a main processor. The addition of a vector processing mode to a slave processor in the system of Bonola would allow the host processor to delegate vector operations to slave processors while continuing to execute other instructions, resulting in greater processing efficiency.

28. Regarding claim 158, Bonola in view of Bridges does not explicitly disclose that a processing resource is a mathematical processing unit.

Mohamed discloses a system similar to that of Bonola in which two processors interact to efficiently process instructions [col. 1, lines 19-26]. The system includes a math co-processor [col. 1, line 24].

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include a mathematical processing unit as a processing resource in the system of Bonola because Mohamed discloses including a math processor in a multi-processor system [col. 1, lines 19-26] and doing so allows the system to more efficiently execute mathematical operations. The use of a mathematical processor as a slave processor in the system of Bonola allows the host processor to be designed for better execution of other types of instructions, allowing for greater overall processing efficiency. Furthermore, the addition of a mode in which a slave processor behaves a mathematical processor allows the system of Bonola to execute mathematical operations, also potentially increasing overall processing efficiency.

29. Regarding claim 1615, Bonola discloses a method of execution-instruction delegation among multiple interdependent processing resources [abstract], the elemental processing resources including a first type and an other type [abstract; host and slave processors], the elemental processing resources being configured to perform processing operations according instructions in an instruction set but being individually incapable of servicing at least one instruction in the instruction set [col. 3, lines 36-51; the slave processor handles instructions that the host processor is unable to], the method comprising:

receiving, from memory, one of multiple execution-instructions of a thread at a first Instruction Processing Unit (IPU) [col. 3, line 26; a command is encountered by a host, or master, processor];

processing the first execution-instruction using the first IPU [col. 3, lines 23-44; if the encountered instruction is not of a type that cannot be executed by the host processor, the host processor continues executing it normally];

receiving an other execution-instruction from of the thread at the first IPU [col. 3, line 26; a command is encountered by a host, or master, processor];

determining that the other execution-instruction from the thread can not be processed by the first IPU [col. 3, lines 25-29; it is determined if the instruction is of a type that cannot be executed by the host processor];

delegating the other execution-instruction from the thread to an elemental processing resources of the other type, the type of the processing resource other than the IPU type processing resource being such that it can process the other execution-instruction from the thread [col. 3, lines 36-51; when an encountered instruction is of a type that cannot be executed by the host processor, the host processor transfers data to a slave processor's registers so that the slave processor may execute the instruction];

maintaining the thread in the first IPU in a sleep state until an indicator that the processing of the other execution-instruction from the thread by the other type processing resource is returned [col. 3, lines 45-57; the thread is suspended in the host while the slave processor executes];

Bonola does not explicitly disclose that the system includes a plurality of master processors that share a plurality of slave processors, wherein instructions of a second thread are executed by a second master processor which can also delegate instructions to a same shared slave processor. However, Bonola does teach that other arrangements of multiprocessor

computer systems could be used for carrying out the invention [col. 7, lines 11-15]. Bridges discloses such an alternate arrangement for a multiprocessor system having multiple master processors and multiple slave processors [Fig. 1]. Bridges further discloses that multiple master processors issue requests to a slave processor [col. 6, lines 33-36]. Such a system allows for any of the master processors to gain the benefit of having a slave processor, and the use of multiple slave processors allows them to do so in parallel, resulting in improved utilization and overall system throughput. It therefore would have been obvious to a person of ordinary skill in the art at the time the invention was made to have multiple host, or master, processors delegate instruction requests to multiple slave processors in the system of Bonola because Bridges teaches such a multiprocessor system for handling master processor requests.

Bonola also does not explicitly disclose that the other type includes an MPU type, an instruction delegating cache type, an encryption or decryption type or a vector type. However, Mohammed discloses a system similar to that of Bonola in which multiple types of processing resources interact to more efficiently execute programs. Mohamed teaches the use of math processors, vector processors, and other types of processors that are able to more efficiently execute specific types of instructions. The use of these processors in a system such as Bonola allows for faster processing of multiple types of instructions and would therefore have been obvious.

Response to Arguments

30. Applicant's arguments filed 11/23/2009 have been fully considered but they are not persuasive. Applicant alleges that none of Bonola, Bridges or Butterworth discloses processors

of a second type. In making these arguments, however, applicant does not reference any claim language or any portion of the prior art references. It is impossible for the examiner to determine on what basis applicant is making these allegations. It is the examiner's position that, as outlined in the rejections above, the master/slave configurations of Bonola, Bridges and Butterworth constitute the claimed invention in that they teach different processing elements that are used to process different types of operations (and wherein instructions are delegated from master processors to slave processors). If applicant disagrees with this position, applicant should state as much and offer supporting evidence in any response to this office action.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Corey Faherty whose telephone number is (571)270-1319. The examiner can normally be reached on weekdays between 7:00 and 4:30, with every other Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272 4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Eddie P Chan/
Supervisory Patent Examiner, Art Unit 2183

/Corey Faherty/
Examiner, Art Unit 2183